

Battlestar Galactica

En un lejano lugar del universo, los *Cylon* han lanzado un ataque imprevisto contra las doce colonias de Kobol. Todo lo que queda de la humanidad es una nave de combate la *Battlestar Galactica* junto con un grupo de naves espaciales civiles. Juntos emprenden un viaje en busca de la decimotercera colonia: la Tierra, cuya ubicación es desconocida. El comandante *Adama* dirige la expedición desde la nave de combate y el resto de las naves se sitúan en línea detrás de la nave principal. El orden en que lo hacen es importante para poder defenderse mejor de los continuos ataques de los *Cylon*.

Después del último ataque las naves han quedado desordenadas y es preciso que recuperen el orden correcto. Sin embargo, el comandante sabe que los *Cylon* preparan un nuevo ataque para dentro de T unidades de tiempo. Si intercambiar el lugar de dos naves consecutivas lleva una unidad de tiempo, calcula si será posible reestructurar la fila antes del ataque, en caso contrario, los pilotos *Apolo* y *Starbuck* tendrán que realizar una salida contra los *Cylon* para intentar retrasar el ataque.

Requisitos de implementación.

El problema se debe resolver de forma recursiva. Indicar el coste obtenido y la recurrencia .

La función que resuelve el problema debe recibir los datos en un vector y devolver el número de unidades de tiempo necesarias para reestructurar la fila de naves.

Entrada

La entrada consta de una serie de casos de prueba. Cada caso consta de dos líneas, en la primera se indica el número de naves que viajan bajo la protección de la *Battlestar Galactica*. En la segunda línea se da la posición en que ha quedado cada nave después del ataque de los *Cylon*. Por ejemplo, si la entrada es 5 2 1 3 4, esto significa que la nave 5 (la que debería viajar en quinto lugar) está ahora en primera posición, la nave que debería estar en la segunda posición esta bien colocada, la nave 1, que debería estar en la primera posición, ocupa la tercera, y las naves que deberían estar en tercera y cuarta posición, han quedado en las posiciones 4 y 5.

Salida

Para cada caso de prueba se escribe en una línea el número de unidades de tiempo necesarias para reestructurar la fila.

Entrada de ejemplo

```
3
3 2 1
6
4 3 2 1 6 5
```

Salida de ejemplo

```
3
7
```

Autor: Isabel Pita.